

## CLAIMS

1        1.        A method of operating a computer, the method comprising:  
2                for data structures in a set of data structures, as unloaded data structures are  
3        needed during runtime,  
4                receiving a data structure from a first memory, the data structure  
5                including one or more sets of instructions and one or more constants;  
6                storing instructions from the data structure in a first portion of a  
7                second memory, the second memory comprising RAM;  
8                storing constants from the data structure in a second portion of the  
9                second memory if only if the respective constant has not been stored in the  
10               second portion of the second memory,  
11               modifying indexes in instructions that reference the constants to  
12               correspond to the respective locations of the constants in the second portion  
13               of the second memory, and  
14               reading and executing at least some instructions from the data  
15               structure from the RAM.

1        2.        The method of claim 1, wherein the data structures comprise classes.

1           3.       The method of claim 1, wherein the data structures comprise Java  
2    classes.

1           4.       The method of claim 1, wherein the sets of instructions comprise  
2    methods.

1           5.       The method of claim 1, wherein the sets of instructions comprise Java  
2    methods.

1           6.       The method of claim 1, wherein the constants from the data structure  
2       comprise a constant pool.

1           7.       The method of claim 1, wherein receiving the data structure from a  
2       first memory comprises receiving the data structure from a server over the Internet.

1           8.       The method of claim 1, wherein modifying indexes in instructions  
2 includes replacing respective indexes with larger indexes and wherein the method  
3 further includes calculating addresses associated with branch instructions.

9. A method of operating a computer, the method comprising:

- for classes in a set of classes, as unloaded classes are needed during runtime,
  - receiving a class from a class file, the class including one or more methods and one or more constants;
  - storing instructions from the class in a first portion of a memory;
  - storing constants from the class in a second portion of the memory if only if the respective constant has not been stored in the second portion of the memory,
  - modifying indexes within methods that reference the constants to correspond to the respective locations of the constants in the second portion of the memory, and
  - executing from the memory at least some instructions from the class before receiving another class from the class file.

10. The method of claim 9, wherein the classes comprise Java classes.

11. The method of claim 9, wherein the memory comprises RAM.

12. The method of claim 9, wherein receiving the class from a class file comprises receiving the class from a server over the Internet.

1           13.     The method of claim 9, wherein modifying indexes within methods  
2 includes replacing respective indexes with larger indexes and wherein the method  
3 further includes calculating addresses associated with branch instructions.

1           14.     The method of claim 9, wherein the respective indexes each comprise  
2 8 bits and the larger indexes each comprise 16 bits.

1           15.     The method of claim 9, wherein the constants comprise strings.

1           ~~16.~~    A computer system comprising:  
2 a memory;  
3 first logic that for classes in a set of classes,  
4 receives a class from a class file, the class including one or more  
5 methods and one or more constants;  
6 stores instructions from the class in a first portion of the memory;  
7 stores constants from the class in a second portion of the memory if  
8 only if the respective constant has not been stored in the second portion of  
9 the memory, and

1 modifies indexes within methods that reference the constants to  
2 correspond to the respective locations of the constants in the second portion  
3 of the memory; and  
4 second logic that executes methods stored in the memory;  
5 wherein the memory, the first logic, and the second logic are coupled locally.

6 17. The computer system of claim 16, wherein the classes comprise Java  
7 classes.

1 18. The computer system of claim 16, wherein the constants from the  
2 class comprise a constant pool of the data structure.

1 19. The computer system of claim 16, wherein the memory comprises  
2 RAM.

1 20. The computer system of claim 16, wherein receiving the class from a  
2 class file comprises receiving the class from a server over the Internet.

1           21.    The computer system of claim 16, wherein modifying indexes within  
2    methods includes replacing respective indexes with larger indexes and wherein the  
3    method further includes calculating addresses associated with branch instructions.

1           22.    The computer system of claim 16, wherein the respective indexes  
2    each comprise 8 bits and the larger indexes each comprise 16 bits.

3           23.    The computer system of claim 16, wherein the logic comprises  
4    computer readable code means loaded into a RAM.